

СОРТУВАННЯ ФАЙЛІВ РОЗМІРОМ БІЛЬШИМ, НІЖ ОПЕРАТИВНА ПАМ'ЯТЬ

Донецький національний університет імені Василя Стуса, м. Вінниця

Сортування – це впорядкування за певною характеристикою ланки об'єктів, що мають конкретні взаємозв'язки, або ж просто мають спільне поле характеристики. Вивчаючи курс АІСД, ми вже ознайомилися із основними методами сортування: бульбашка, вибором, вставками, пірамідальним, швидким, злиттям. Усі ці алгоритми можна класифікувати як алгоритми внутрішнього сортування, тобто ті, у яких дані зберігаються у оперативній пам'яті комп'ютера, і якісь маніпуляції ми проводимо тільки з ними. На противагу алгоритмам внутрішнього сортування, існують алгоритми зовнішнього сортування – це спосіб сортування даних, які не вміщуються в оперативну пам'ять та є розташованими на периферійних пристроях. Важливою характеристикою є факт того, що внутрішнє сортування значно ефективніше зовнішнього, оскільки на звернення до оперативної пам'яті комп'ютера витрачається набагато менше часу, ніж до периферійних пристроїв. Тобто, до прикладу, відсортувати масив із певною кількістю елементів, що знаходиться в оперативній пам'яті в рази простіше, ніж аналогічний масив, що записаний на Ufile. Простими словами, зовнішнім сортуванням ми будемо впорядковувати файли, що мають розмір значно більший за середньостатистичний розмір сучасних оперативок – файли розміром в Гігабайти.

Впорядкована підпоследовність файлу – саме таким є визначення основного поняття зовнішнього сортування, а саме відрізка (серії). Елементи файлу діляться на вже впорядковані частини (відрізки): чим краще впорядкований умовний масив, тим менше відрізків утвориться, відповідно, сортування відбудеться швидше. Саме сортування складається із последовності кроків, в кожному з яких розподіляється вміст файлу у два підфайли: в перший – з непарним індексом, в другий – з парним. Вже потім їх злиття записується у сам файл. Це був один з основних методів, відомий як пряме злиття. Окрім нього ще існує природне (адаптивне), збалансоване багатошляхове злиття та багатофазне сортування (Фібоначчі).

В будь-якому випадку, який би метод не вибрали, ми завжди матимемо один нюанс. Якщо просто реалізувати алгоритм зовнішнього сортування, то зрозуміло, що програма колись відсортує файл. Постає питання: а коли саме? Це може тривати навіть декілька років, якщо, припустимо, розмір файлу є 20 Гігабайтів. Такий час виконання є просто неприпустимим. Тому для покращення ефективності зовнішнього сортування, необхідно додатково ще працювати із оперативною пам'яттю. А саме, до початку сортування, заснованого на

застосуванні серій, початковий файл частинами має зчитуватись в оперативну пам'ять, до кожної частини застосовується один з найбільш ефективних алгоритмів внутрішнього сортування і відсортовані частини, що утворюють серії, записуються в новий файл. Пророблені маніпуляції пришвидшать роботу, умовно до декількох днів, але це всеодно ще є неприйнятним для нас. Задля очікуваного результату необхідно використати наступне. Використати буфер, або як його ще називають: відображення файлу в пам'ять. Це спосіб роботи з файлами в деяких операційних системах (усі сучасні це підтримують), при якому всьому файлу або певній безперервній його частині ставиться у відповідність певна ділянка пам'яті (діапазон адрес оперативної пам'яті). При цьому читання даних із цих адрес фактично призводить до читання даних з відображеного файлу, а запис даних за цими адресами призводить до запису цих даних у файл [1]. Тобто ми вже будемо читати не щоелементно (як у с++), а фактично працювати із самим файлом як із масивом (або структурою даних), що знаходиться у оперативній пам'яті. До речі, деякі сучасні мови програмування (с# наприклад) вже мають свій вбудований механізм відображення в пам'ять. Це значно полегшує роботу із доволі великими файлами.

Отже, сортування файлів розміром більшим, ніж оперативна пам'ять можливо та доволі не складно. Зазвичай, ми не задумуємось над таким питанням, або вважаємо, що це не можливо. Проте, навіть попри великий спектр алгоритмів, слід завжди пам'ятати також про постійне залучення оперативної пам'яті та буфер, навіть якщо він вбудований у середовище із яким працюємо, бо тільки у такий спосіб ми зможемо отримувати дані раціонально та ефективно.

Список літератури

1. Головченко М. М. *Проектування алгоритмів//Курс лекцій для студентів спеціальності 121//Тема 1 – Алгоритми Сортування – с. 7-18.*
2. *Зовнішнє сортування / [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/%D0%97%D0%BE%D0%B2%D0%BD%D1%96%D1%88%D0%BD%D1%94_%D1%81%D0%BE%D1%80%D1%82%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F*